(54) Title: SYSTEM AND METHOD FOR PROVIDING CALL CENTER–BASED CUSTOMER SERVICES

(57) Abstract

A system and method for providing call center services, such as customer surveys and order entry, is data-driven and can be used to perform services for many different customers, without developing a separate application program for each customer. A product, running as an application on a PC workstation, provides basic call flows that are easily customized for specific customers by integrating customer profiles with an underlying application. The underlying application retrieves pre-defined customer profiles for various services at start up. When an agent workstation receives a call, the application references the particular customer profile by a key, such as the called number. The application then performs one of several basic call flow processes, such as a question and answer survey, taking orders for a customer's product, etc., under the customer profile. The application reads and interprets the customer profile, and inserts information specific to that customer into specific places during the call process.
FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>AL</td>
<td>Albania</td>
<td>ES</td>
<td>Spain</td>
<td>LS</td>
<td>Lesotho</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AM</td>
<td>Armenia</td>
<td>FI</td>
<td>Finland</td>
<td>LT</td>
<td>Lithuania</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AT</td>
<td>Austria</td>
<td>FR</td>
<td>France</td>
<td>LU</td>
<td>Luxembourg</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AU</td>
<td>Australia</td>
<td>GB</td>
<td>United Kingdom</td>
<td>LV</td>
<td>Latvia</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AZ</td>
<td>Azerbaijan</td>
<td>GE</td>
<td>Georgia</td>
<td>MC</td>
<td>Monaco</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BA</td>
<td>Bosnia and Herzegovina</td>
<td>GE</td>
<td>Georgia</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BB</td>
<td>Barbados</td>
<td>GH</td>
<td>Ghana</td>
<td>MD</td>
<td>Republic of Moldova</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BE</td>
<td>Belgium</td>
<td>GN</td>
<td>Guinea</td>
<td>MG</td>
<td>Madagascar</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BF</td>
<td>Burkina Faso</td>
<td>GR</td>
<td>Greece</td>
<td>MK</td>
<td>The former Yugoslav</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BG</td>
<td>Bulgaria</td>
<td>HU</td>
<td>Hungary</td>
<td>ML</td>
<td>Mali</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BI</td>
<td>Benin</td>
<td>IE</td>
<td>Ireland</td>
<td>MN</td>
<td>Mongolia</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BR</td>
<td>Brazil</td>
<td>IL</td>
<td>Israel</td>
<td>MR</td>
<td>Mauritania</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BY</td>
<td>Belarus</td>
<td>IS</td>
<td>Iceland</td>
<td>MW</td>
<td>Malawi</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CA</td>
<td>Canada</td>
<td>IT</td>
<td>Italy</td>
<td>MX</td>
<td>Mexico</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CF</td>
<td>Central African Republic</td>
<td>JP</td>
<td>Japan</td>
<td>NE</td>
<td>Niger</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CG</td>
<td>Congo</td>
<td>KE</td>
<td>Kenya</td>
<td>NL</td>
<td>Netherlands</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CH</td>
<td>Switzerland</td>
<td>KG</td>
<td>Kyrgyzstan</td>
<td>NO</td>
<td>Norway</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CI</td>
<td>Côte d’Ivoire</td>
<td>KP</td>
<td>Democratic People’s Republic of Korea</td>
<td>NZ</td>
<td>New Zealand</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CM</td>
<td>Cameroon</td>
<td>KR</td>
<td>Republic of Korea</td>
<td>PL</td>
<td>Poland</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CN</td>
<td>China</td>
<td>KZ</td>
<td>Kazakhstan</td>
<td>PT</td>
<td>Portugal</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CU</td>
<td>Cuba</td>
<td>LC</td>
<td>Saint Lucia</td>
<td>RO</td>
<td>Romania</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CZ</td>
<td>Czech Republic</td>
<td>LI</td>
<td>Liechtenstein</td>
<td>RU</td>
<td>Russian Federation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DE</td>
<td>Germany</td>
<td>LK</td>
<td>Sri Lanka</td>
<td>SD</td>
<td>Sudan</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DK</td>
<td>Denmark</td>
<td>LR</td>
<td>Liberia</td>
<td>SE</td>
<td>Sweden</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EE</td>
<td>Estonia</td>
<td>SI</td>
<td>Slovenia</td>
<td>SG</td>
<td>Singapore</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SK</td>
<td>Slovakia</td>
<td>SN</td>
<td>Senegal</td>
<td>SZ</td>
<td>Swaziland</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TD</td>
<td>Chad</td>
<td>TG</td>
<td>Togo</td>
<td>TJ</td>
<td>Tajikistan</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TM</td>
<td>Turkmenistan</td>
<td>TR</td>
<td>Turkey</td>
<td>TT</td>
<td>Trinidad and Tobago</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UA</td>
<td>Ukraine</td>
<td>TG</td>
<td>Uganda</td>
<td>US</td>
<td>United States of America</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UG</td>
<td>Uganda</td>
<td>VI</td>
<td>United States of America</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UZ</td>
<td>Uzbekistan</td>
<td>VN</td>
<td>Viet Nam</td>
<td>YU</td>
<td>Yugoslavia</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ZW</td>
<td>Zimbabwe</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SYSTEM AND METHOD FOR PROVIDING CALL CENTER-BASED CUSTOMER SERVICES

TECHNICAL FIELD

The present invention relates generally to telecommunication systems and, more particularly, to a system and method for handling calls and collecting data therefrom.

BACKGROUND OF THE INVENTION

A call center is commonly used to provide customer services, such as caller surveys, telemarketing, order entry, technical assistance, etc. A typical call center consists of multiple agent consoles, where each console is operated by a human agent to provide customer services to incoming callers. A call center also includes a system for receiving and distributing telephone calls to the agent consoles and other intelligent peripherals, such as audio response units.

Each agent console typically consists of a personal computer (PC) workstation, a phone pad for inputting numeric or DTMF input, and a head set. A specific software application running on the PC workstation performs a series of steps or "call flow" to guide the agent through the handling of the call. The specific software application, for example, provides scripts to the agent that present textural questions, displayed to the agent, which the agent asks the caller. The specific software application also includes functions for entering the caller's responses to the questions, taking orders from the caller, and providing the agent with series of options, and suboptions, under predefined decision trees. The software application defined in the call flow also includes certain automated functions to process incoming calls, such as creating billing records, interacting with an automated call distributor, and printing files containing information collected during a call.

Many companies that provide call center services must also provide the same services for many different customers. For example, one customer may run a mail order catalog company, while another customer may request customer survey
information based on a current advertising campaign. As a result, the company providing call center services must develop customized software applications for each customer. Additionally, the agents operating the agent workstations must learn each new application for each new customer.

5 SUMMARY OF THE INVENTION

An embodiment of the present invention includes a telecommunication system for receiving and processing calls, which includes a server and at least one call receiving station. The server stores a plurality of profiles specified by users, wherein each profile specifies details of a specific call flow process. The call receiving station is coupled to the server and coupled to receive incoming telephone calls. The call receiving station retrieves one of the plurality of profiles and defines a call flow process based on the retrieved profile and an executable routine. The defined call flow process provides questions for an incoming telephone call. The call receiving station collects answers to the questions.

Additionally, an embodiment of the present invention includes a computer-implemented method for use in a telecommunication system. The method includes the steps of: (a) retrieving one of a plurality of stored profiles, each profile containing call flow data specified by users; and (b) defining a call flow process based on the retrieved profile and an executable routine, wherein the call flow process provides questions for a call receiver that services telephone calls.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram showing a system configuration suitable for practicing an exemplary embodiment of the present invention.

Figure 2 is a block diagram showing a logical architecture of a portion of the system of Figure 1.

Figure 3 is a data flow diagram for data collected and routed under the system of Figure 1.
Figure 4 is a block diagram of an exemplary agent console shown in the system of Figure 1.

Figure 5 is a block diagram of a logical architecture of the agent console shown in Figure 1.

Figure 6 is a flow diagram showing steps performed by at least a portion of the system of Figure 1.

Figure 7 is a flow diagram showing steps performed under a call flow application step of Figure 6.

Figure 8 is a flow diagram showing steps performed by a customer service step of Figure 7.

Figure 9 is a front view of a computer screen showing an exemplary screen display on the agent console.

Figure 10 is a front view of the computer screen showing an exemplary order entry screen display on the agent console.

DETAILED DESCRIPTION OF THE INVENTION

A computerized network, and in particular, a method and apparatus for providing call center-based customer response services in a telecommunications network, is described in detail herein. In the following description, numerous specific details are set forth such as data formats, user interface screens, call flows, menu options, etc., in order to provide a thorough understanding of the present invention. One skilled in the relevant art, however, will readily recognize that the present invention can be practiced without some of the specific details described herein, or with other specific details. Well-known structures and steps are not shown or described in detail in order to avoid obscuring the present invention.

1. Overview

A system and method for providing call center services, such as customer surveys and order entry, is data-driven and can be used to perform services for many different customers, without developing a separate application program for each
customer. A product, running as an application on a PC workstation, provides basic call flows that are easily customized for specific customers by integrating customer profiles with an underlying generic application. The underlying application retrieves predefined customer profiles for various services at start up. The profile specifies other features, such as lists of products and services offered by the customer for which orders can be taken, questions to ask during the course of a caller survey, the customer’s business name to use, and data formats and processes for entering orders or data. The terms “profile” and “customer profile” are generally used interchangeable herein.

When an agent workstation receives a call, the application references the particular customer profile by a key, such as the called number (e.g., the customer’s 1-800 number). The application then performs one of several basic call flow processes, such as a question and answer survey, taking orders for a customer’s product, etc., under the customer profile. The application reads and interprets the customer profile, and inserts information specific to that customer into specific places during the call process. For example, the application displays a textual script for the agent at the agent console to use as an initial greeting for the caller; the script is customized for each customer by specifying the text in the customer’s profile.

The exemplary embodiment allows a call service provider to provide call center services for many different customers without developing a separate, costly software application for each customer. The exemplary embodiment defines basic call flows and data fields to be used in any customer application and the ability to readily customize a call flow for each customer under data-driven customer profiles. Customer profiles can be generated by a non-programmer, such as a member of a sales or marketing account team. The exemplary embodiment, based on the profile, automatically generates a software application specific to the customer. Each automatically generated customer application has substantially the same user interface, or look and feel. Thus, agents at agent consoles need not learn a new application program for each customer.
2. **System Description**

As used generally herein, a "service provider" is a company that performs call center services for other companies or customers. A "customer" typically refers to a company for whom services are being performed by the service provider. A customer typically owns one or more telephone numbers such as toll free numbers (1-800 or 1-888 numbers), and contracts a service provider to handle calls for at least one of the numbers. A "caller" typically refers to a person calling the number, and who is routed to the service provider's call center. A caller is typically, but not necessarily, a customer/client/prospect of the above-defined "customer."

Figure 1 depicts an environment that is suitable for practicing the preferred embodiment of the present invention. This environment includes a call center 105 that may be accessed by caller 100 via a public switched telephone network (PSTN) 102. The PSTN 102 may be connected to the call center 105 via a voice release trunk (RLT) 104. The call center 105 includes an automatic call distributor (ACD) 106 for distributing calls within the call center. The ACD is connected via a voice trunk 110 to an automated response unit (ARU) 108. The ARU 108 provides voice response and menu routing functions to a caller.

The call center 105 may also include an application processor (AP) 114 that is associated with the ACD 106. The AP 114 may be a dedicated computer system that provides intelligent application processing for assisting the ACD 106. In particular, certain functionality that may be performed by the ACD 106 is off loaded to the AP 114 to enable the ACD to focus on performing switching and queuing functions. The AP 114 is linked to the ACD 106 via an ISDN implementation of a switch/computer application interface (SCAI) link 112.

The call center 105 may include multiple agent consoles 118. These consoles are connected to the ACD 106 via voice trunks 116. Each agent console may include a workstation, a telephone pad, and a headset. Human operators at the agent consoles perform scripts, prompting, and transferring of calls that are needed for call processing on behalf of customers.
The call center 105 may also include multiple network information distributed services (NIDS) 120. Each of the NIDS 120 may be implemented as a separate computer system. The NIDS 120 may be redundant, and generally serve the role of storing database information, including data and billing records. The NIDS 120 may all be connected to an Ethernet local area network (LAN) 42 that also interconnects the ARU 108, the AP 114, the agent consoles 118, a boot server 124, and a validation server 126.

The boot server 124 stores the customer profiles in the call center 105 and is responsible for distributing the customer profiles to other entities within the call center 105. The validation server 126 is responsible for performing credit card validations for caller orders. The validation server 126 may be connected via a network, such as an X.25 network 138 to a credit card validation system 140.

The boot server 124 is connected via a LAN or wide area network (WAN) 130 to an answernet delivery service (ADS) 132. The ADS 132 may be implemented by a midrange computer system that is responsible for delivering customer profiles to the appropriate boot servers via the LAN/WAN 130. The ADS 132 may be connected to multiple call centers and may distribute the customer profiles to the appropriate call centers. Customer profiles are received from the profile editor 136. The customer profiles may be emailed over an email network (such as a PSTN, the internet, an intranet, or a LAN/WAN) 134 to the ADS 132. The profile editor 136 may be run on a personal computer, such as a portable computer that may be carried by a marketing representative, as has been discussed above.

Figure 2 illustrates the flow of a customer profile from the profile editor 136 to the other components within the call center. The profile editor 136 emails (see arrow 180 in Figure 2) the customer profile to the ADS 132. The ADS sends the customer profile over a LAN/WAN 130 to one or more boot servers. In the example shown in Figure 2, the customer profile is sent to boot servers 124A, 124B, and 124C. The boot servers 124A, 124B, and 124C then distribute the customer profiles to applications running on agent consoles 118 as needed. In the examples shown on Figure 2, the boot server 124A distributes customer profiles over a LAN 122 to three
agent consoles 118. Application programs running on the agent consoles utilize the
customer profiles to customize the handling of calls on behalf of the customers. These
applications compile data collected from the caller into records that are sent over LAN
122 to an NIDS server 120. These records are added to a database maintained on the
NIDS server 120.

Figure 3 illustrates the flow of collected data in profiles throughout the
call center. A caller 100 provides information to an operator over a voice link or an
operator console 118. Programs on the operator console 118 generate records in
response to the call, as discussed above, which are sent to the NIDS server 120. The
NIDS server 120 may be connected to a mainframe computer system 182 that is
responsible for maintaining a large database of call center information such as call
response and billing data.

Referring to Figure 4, each operator console 118 includes a central
processing unit (CPU), such as an Intel 486 or Pentium processor, that manages and
operates functions of the console. The operator console 118 may also include a number
of peripheral devices, including a keyboard 68, a mouse 70, and a display device 72,
such as a video display device. The agent console 118 may have a modem 74 for
interfacing with the telephone lines and a network adapter 76 for interfacing with the
Ethernet LAN. A telephone keypad 84 and head set 86 allow an agent operating the
agent console 118 to send and receive telephone calls, as is typical for call center agent
stations.

The agent console 118 includes primary storage 78 and/or secondary
storage 80. The primary storage 78 includes an operating system 82, such as the IBM®
OS/2® operating system. The primary storage 78 may also include executable and
configuration files, described below. The secondary storage 80 may hold a number of
different types of data or programs, including the operating system. Those skilled in the
art will appreciate that the computer architecture depicted in Figure 4 is intended to be
merely illustrative and not limiting the present invention. The present invention may be
practiced with other computer architectures, including distributed systems.
Referring to Figure 5, a logical architecture of the agent console 118 includes three executable files CONSOLE.EXE 150, MOC.EXE 151 and GUI.EXE 152. CONSOLE.EXE 150 performs core processing of call flows in the agent console 118, while GUI.EXE 152 provides a graphical user interface (GUI) for the agent for display on the display device 72, as described more thoroughly below. MOC.EXE 151 transfers and receives data to and from the agent console 118, as described below.

A communication interface process 154 communicates with a similar communication interface process 156 in the boot server 124. The communication interface process 154 also communicates with the NIDS server 120, as well as other components connected to the Ethernet LAN 122. A configuration file, MOC.DLT 158, contains information such as which profiles the agent console 118 should retrieve from the boot server 124. In the exemplary embodiment, MOC.DLT 158 identifies directories or locations of profiles on one or more of the boot servers 124 of profiles to be downloaded to the agent console 118 (e.g., multiple profiles may reside in a given directory). In the exemplary embodiments, MOC.DLT 158 and MOC.EXE 151 may employ wildcard characters to represent more than one character and thereby retrieve multiple profiles. Based on the configuration file MOC.DLT 158, MOC.EXE 151, via the communication interfaces 154 and 156, retrieves an appropriate profile from a profile database 160 in the boot server 124, and stores it in local storage, such as a local database PROFILE.DB 162 in the agent console 118.

Referring to Figure 6, an exemplary routine 500 for creating customer profiles and processing calls under such profiles is shown. In step 502, a sales or marketing representative creates a customer profile. In the exemplary embodiment, the representative creates the customer profile using a profile editor, such as that described in detail in copending U.S. patent application No. ________________, entitled “System For Compiling and Integrating Customer Profiles For Call Center Services,” filed concurrently herewith and assigned to the same assignee as the present application.

In step 504, the representative uploads the customer profile to the ADS 132 via an e-mail network, such as a PSTN, Internet, Intranet, LAN/WAN, etc. In
the exemplary embodiment, the representative attaches the profile to an e-mail sent to the ADS 132. The ADS 132 receives the e-mail, extracts the customer profile, and validates the profile, as discussed above. If the profile meets the validation requirements, then in step 506, the ADS 132 distributes the profile to one or more of the boot servers 124. The ADS 132 uses an internal configuration file to determine which profiles to send to which of the boot servers 124. If the profile is not validated, then the ADS 132 returns the message to the representative indicating that the profile was not validated, preferably with a reason code to help instruct the representative as to how to correct the error.

In step 508, the agent console 118 retrieves one or more customer profiles from one or more of the boot servers 124. When the agent console 118 is initialized, or powered up, MOC.EXE 151 queries MOC.DLT 158 to determine which customers will be handled by the agent console 118. Customers are typically identified by their 800/888 toll-free number ("800 number") that callers may dial.

Each customer profile stored on the boot servers 160 is stored as a file named for or with the customer’s 800 number. Each profile also includes an activation date and time. The activation date/time represents the date and time that the profile will become activated. As a result, representatives can create the profile for future use, and upload and store it in one of the boot servers 160, without the profile being used and prior to its activation date/time.

At start up, MOC.EXE 151 determines from MOC.DLT 158 which profiles to retrieve from the boot server 124. MOC.EXE 151 then places these retrieved profiles in the profile database PROFILE.DB 162. When the ACD 106 receives a call, the ACD sends a call offer message containing the dialed number to the AP 114. The AP 114 then forwards the call to an appropriate agent console 118 previously specified as handling calls for the dialed number. Thereafter, CONSOLE.EXE 150 uses the dialed number as a key to retrieve the appropriate customer’s profile from PROFILE.DB 162. Alternatively, multiple numbers can be associated with the same customer profile. Therefore, CONSOLE.EXE 150 retrieves a property record based on
the dialed number, where the property record identifies the appropriate customer profile, and then retrieves the appropriate customer profile.

In an alternative embodiment, at start up and after determining from MOC.DLT which profiles to retrieve, MOC.EXE 151 queries the profile database 160 on the boot server 124 and retrieves the profiles specified in MOC.DLT. MOC.EXE 151 then places the retrieved profiles in the local database PROFILE.DB 162. When the agent console 118 receives a call, CONSOLE.EXE 150 retrieves from the database PROFILE.DB 162 only the profile named or corresponding to the dialed number that has the most recent activation date/time, and which is not in the future. In yet another alternative embodiment, certain agent consoles 118 provide call center services for particular customers. Therefore, PROFILE.DB 162 locally stores certain customer profiles therein. MOC.EXE 151 first determines whether the appropriate customer profiles are stored in the local profile database PROFILE.DB 162. MOC.EXE 151 then queries the profile database 160 on the boot server 124 via LAN messaging and the communication interfaces 154 and 156 to determine if any new profiles exist for the given 800 number.

In step 510, the agent console 118 receives a call and performs a call flow process to handle the call. CONSOLE.EXE 150 loads from PROFILE.DB 162 the customer profile associated with the received call and performs a call flow defined generically by compiled source code customized by the customer profile. Rather than a compilation of source code, the call flow process or application is an interpretation by executable code. CONSOLE.EXE 150 reads data in from the customer profile stored in the profile database PROFILE.DB 162, and interprets that data to perform the call flow process. This process is much quicker in execution than compiling customer flow data with source code at one time during a call. Additionally, this process is much quicker to develop than developing individual applications, customized for each customer. An exemplary call flow process is described in more detail below with respect to Figure 7.

In step 512, the agent console 118 writes data collected from the caller to a database on the NIDS server 120. More specifically, when the call is completed, CONSOLE.EXE 150 formats the collected data to customer specifications identified in
the customer profile, and then writes the data to a results file in the NIDS 120. Alternatively, the agent console 118 may locally store the collected data if the agent console determines that it cannot communicate with the NIDS 120. The customer specifications identified in the profile can include how many columns are defined, if delimiters are to be used (e.g., commas), what type of delimiters are to be used, etc. CONSOLE.EXE 150 sends the collected and formatted data to the NIDS server 120 as individual data records. Each record identifies the customer, typically by 800 number and date/time. Thus, the collected data can be available to the customer within minutes of being collected.

In step 514, CONSOLE.EXE 150 creates a billing data record ("BDR") with call statistics, such as the caller's ANI, dialed number, customer identifier, call duration, call date/time, agent that handled the call, services provided to the caller, etc. CONSOLE.EXE 150 sends the BDR to the NIDS server 120. The BDR will be forwarded by the NIDS to downstream call processing billing systems, for billing purposes. The NIDS server 120 also sends the BDR to the customer data delivery system 128 so that other types of customer reports can be provided to the customer, or to the call service provider. Such reports can include total number of calls received, total number of calls handled, average call handling time, average call waiting time, etc.

In the exemplary embodiment, two types of BDR-based reports are provided: a summary report on statistics of all calls handled during a time period, and an individual call detail report.

The agent console 118, when creating a BDR record, uses data generated by the agent console, as well as data received from the ACD 106. CONSOLE.EXE 150 of the agent console 118 performs the BDR creation function, which has many common elements or fields, regardless of the individual customer application. Therefore, a separate application need not be developed for the agent console 118 for each customer profile to create billing records. Instead, the representative may create any customized BDR when creating the customer profile. Overall, CONSOLE.EXE 150 and GUI.EXE 152 perform at least steps 508 through 514 of the routine 500.
Referring to Figure 7, an exemplary call flow process for handling an incoming call under a previously defined profile will now be described. In steps 550 and 552, CONSOLE.EXE 150 and GUI.EXE 152 read in the retrieved profile data and populate or display fields in a display window on the display device 72 with associated text. For example, each call flow generally begins with a greeting and survey-type process. CONSOLE.EXE 150 determines from the profile that a particular greeting is to be displayed, retrieves from the customer profile a text field containing the greeting, and triggers GUI.EXE 152 to display such text. GUI.EXE 152 displays the textual greeting for the agent to read. CONSOLE.EXE 150 also determines from the profile which questions the agent should ask the caller by retrieving from the customer profile fields that contain the number of questions to be asked and associated questions, and triggers GUI.EXE 152 to display them. The agent can then see how many questions are to be asked, and proceed to ask them. Exemplary display windows or screens are described below with respect to Figures 9 and 10.

CONSOLE.EXE 150 also retrieves from the customer profile fields that specify allowed answers (i.e., “yes” or “no,” integer values, alphanumeric text, etc.), field formats for entering the answers, indicators as to whether each field is required or optional, etc. CONSOLE.EXE 150 provides such fields, field formats, indicators, etc. to GUI.EXE 152. When the agent enters the caller’s answers, GUI.EXE 152 validates these answers (answer type, answer format, required fields to be populated, etc.) and records them in the proper fields. For example, most call flow processes require four fields: caller’s name (alphanumeric text), complete address (alphanumeric text), daytime phone number (integer values) and evening phone number (integer values). GUI.EXE 152 displays such fields in similar locations on the screen so that the agent always interacts with a common interface, regardless of the call flow process. Additional optional fields can also be provided in the customer profile and displayed on the screen for input by the agent. In general, CONSOLE.EXE 150 and GUI.EXE 152 provide certain generic text and display window items for call flow processes, while the customer profile provides specific product tables, options and scripts.
Following steps 550 and 552, the call flow process branches depending upon the type of call flow process being performed. In the exemplary embodiment, three types of basic call flow processes are defined: direct response, survey and order entry. Direct response and survey call flow processes are question and answer sessions, in which the agent asks the caller a series of questions, and records the caller’s answers in fields defined by the customer profile. The customer profile for direct response and survey call flow processes include a greeting script (as noted above), the number of questions to ask, the questions themselves, allowed answers, answer format, and data format for delivery to the customer. The customer profile for direct response and survey call flow processes may also include items or fields such as customer name, address, delivery options, billing methods, and other information needed to provide the service specified by the customer.

Examples of direct response call flow processes or applications include creating a database of a customer’s clients/prospects, compiling “change of address” information, enrolling callers into a program offered by the customer, tracking responses to a particular advertising campaign or program, or fulfilling a caller’s request for information, e.g., request for product brochures. As a result, a direct response call flow process is typically a short session, generally consisting of five to ten questions. A survey call flow process is more extensive. It provides branching capabilities in which the answers to questions determine further questions and processing. Examples of survey call flow processes include obtaining client/prospect survey information, allowing callers to register products they purchased, allowing more “open ended” responses, opinions, suggestions, and allowing the agent to collect more specific details from questions.

Order entry call flow processes are used for placing caller’s orders for a customer’s products and/or services. Order entry call flow processes provide a greeting and other scripts for the agent, as well as tables of products and services offered, and lists options for each product selected. Thus, order entry call flow processes include not only the four required data fields for direct response/survey, but also three additional required fields: credit card number (integer value), credit card expiration date (integer
value) and credit card authorization code (integer or alphanumeric text). Order entry call flow processes may also include additional fields, such as caller account number, fields for entering orders, fields to provide additional pieces of information needed to define or describe the type of item ordered, and/or full descriptions of the companies’ products/services. The caller account number is used for callers who have previously ordered merchandise, and thus the name, address and phone number fields are already available. Thus, CONSOLE.EXE 150 retrieves from the boot server or the local database 162, data for such fields. Examples of order entry call flow processes include college registration, ordering event tickets, home shopping television networks, catalog shopping, etc. The agent console 118 may receive additional data directly from the customer, such as caller account numbers, etc.

In step 554, CONSOLE.EXE 150 determines whether the customer profile corresponds to an order entry call flow process. If so, then CONSOLE.EXE 150 in step 556 displays and enables an order entry soft key. The order entry soft key is displayed in the display window and indicates a key on the keyboard 68 that may be pressed to place an order. When this key is selected by the agent, the order entry screen appears for taking orders, validating credit cards, etc., as described below.

Following steps 554 and 556, CONSOLE.EXE 150 and GUI.EXE 152 in step 558 collect data from the caller based on input from the agent. Much of the data collected in step 558 is common to all three call flow processes (i.e., direct response, survey and order entry). Thus, the agent console 118 displays a greeting and initial questions for the agent to read to the caller and fields for the agent to input responses from the questions. For direct response or survey processes, CONSOLE.EXE 150 and GUI.EXE 152 display the questions in sequence for the agent to ask, or display all questions at once and let the agent select which questions to ask in what sequence.

In step 560, CONSOLE.EXE 150 determines whether certain keys have been pressed. If the order entry soft key has been depressed, then in step 562, CONSOLE.EXE 150 determines whether required data in corresponding data fields have been entered. For example, the order entry call flow process may require an initial credit card validation process. The call flow process can only display products and
services after a credit card is validated, thus reducing call time for callers whose cards cannot be validated. If the credit card data has not been entered, then the call flow process loops back to steps 558 and 560.

If credit card data is entered in step 560, then in step 564, the agent console 118 transmits the credit card information to an appropriate credit card validation center or the validation server 126. In step 566, CONSOLE.EXE 150 determines whether the credit card data was validated. If not, CONSOLE.EXE 150 and GUI.EXE 152 sets a credit card field status to an error value in step 568, and displays an appropriate error script for the agent in step 570. If the credit card data is valid in step 556, or following step 570, the call flow process loops back to steps 558 and 560.

Assuming the credit card data was entered and validated, then following step 562, CONSOLE.EXE 150 and GUI.EXE 152 in step 572 collect order information from the caller. During the order entry call flow process, the agent under CONSOLE.EXE 150 can simply enter an item number at any time during the process to access another screen which displays detailed information about that item. In response to the input item number and an appropriate key depressed, CONSOLE.EXE 150 and GUI.EXE 152, based on data in the customer profile, displays detailed information about the item in the display window. For example, a caller may want to know more details about an item before actually purchasing that item. The operator then selects the item number to access an “item number/product description” screen and provide the caller with the appropriate information over the phone. In step 574, CONSOLE.EXE 150 determines whether a survey key or done key has been depressed by the agent. If the done key has been depressed, then in step 576, CONSOLE.EXE 150 determines whether all required data has been entered. If not, the call flow process returns back to steps 558 and 560. Similarly, if the survey key has been depressed, indicating that additional information is requested from the caller based on additional questions, the call flow process loops back to steps 558 and 560.

If the done key is depressed in step 560, and/or all required data is entered in step 576, then in step 578, CONSOLE.EXE 150 determines whether the customer profile and call flow process correspond to an order entry process. If not, then
in step 580, CONSOLE.EXE 150 generates a BDR, as noted above. Thereafter, in step
582, CONSOLE.EXE 150 writes or creates a response for the caller and/or a report for
the customer. A response for the caller may include faxing confirmation of an order or
requesting information. In the exemplary embodiment, the agent console 118, via the
modem 47, faxes documents stored in the secondary storage 80 or stored elsewhere.
For example, the customer may be advertising an 800 number to call and receive an
immediate and complete listing of the customer’s products and services along with how
they compare to the competition. The caller dials the 800 number and requests the
document from the agent. The agent simply fills the name and address information,
together with fax number and possibly other data in the appropriate fields of the display
window under the above steps, and faxes the document to the caller. Alternatively, the
agent console 118 may transfer calls to another platform to either automatically request
that the other platform fax or transmit the appropriate information to the callers (with or
without transferring calls), or allow the callers themselves to obtain the information by
interacting with the other platform.

In step 584, CONSOLE.EXE 150 and GUI.EXE 152 read from the
customer profile and display on the display window a closing script. The agent reads
the closing script and ends the call.

If CONSOLE.EXE 150 determines in step 578 that an order entry call
flow process is being performed, then in step 586, the agent console 118 calculates a tax
for the order based on the State in which the caller resides, whether goods or services
are purchased, the particular goods/services purchased, etc. In the exemplary
embodiment, the agent console 118 communicates with a tax server to obtain the
appropriate tax information for each order.

In step 588, CONSOLE.EXE 150 and GUI.EXE 152 displays the order
entry totals in the display window. In step 590, CONSOLE.EXE determines whether a
modify order key has been selected by the agent. If so, then in step 590, the agent
collects additional order information from the caller, and the call flow process loops
back to step 574.
If CONSOLE.EXE 150 determines in step 590 that the process order key has been selected, then in step 594, the agent console 118 validates and posts the credit card data with the order amount. In step 596, CONSOLE.EXE 150 determines whether the credit card data with the amount is valid, and if not, sets the credit card field status to an error value and sets and displays an appropriate error script in steps 598 and 599, in a manner similar to that under steps 568 and 570. If the credit card data and amount are valid, then the call flow process performs steps 580-584 as described above and the call ends.

During the call flow process, CONSOLE.EXE 150 provides the ability for the agent to “take back and transfer” the incoming call. Thus, the agent can transfer the call to another 800 number upon request of the caller or at the option of the agent, at any time during the call. Callers often request to be connected with a customer service center. Alternatively, a caller may have just ordered various items from the agent and is now requesting to be transferred to the customer’s billing department. The agent can simply out-dial the appropriate number to transfer the caller to the customer.

In the exemplary embodiment, CONSOLE.EXE 150 and GUI.EXE 152 provide a customer service soft key displayed on the screen for the agent. In step 560, CONSOLE.EXE 150 determines whether the agent has selected the customer service key. If so, then a customer service subroutine 600 is invoked.

Referring to Figure 8, if the customer service key is depressed in step 560, then CONSOLE.EXE 150 determines in step 601 of the customer service subroutine 600 if the current time is within the range of business hours for the customer (based on the customer’s customer service number (“CS#”)). If so, then CONSOLE.EXE 150 and GUI.EXE 152 records data collected from the caller, such as name, address and phone number for the caller in step 602. The data recorded can also include how the call was terminated. In step 604, CONSOLE.EXE 150 writes a BDR, as noted above. In step 606, the agent console 118 completes and transfers the call to the customer’s customer service number. In the exemplary embodiment, the agent console 118 provides conference call functionality to permit the agent, caller and a customer service representative to simultaneously conference during transfer of the call.
The agent console 118 may also deliver certain data to the customer service center such as the data previously collected.

If the current time is outside the business hours for the customer's customer service center (as defined in the customer's profile), then in step 608, CONSOLE.EXE 150 and GUI.EXE 152 provide a textual script in the display window for the agent to inform the caller of the business hours of the customer's customer service center, provide them with a customer service telephone number for the customer, and offer to help them further. In step 610, CONSOLE.EXE 150 determines whether the caller requires further assistance based on a further assistance key depressed by the agent. If so, then in step 612, the customer service process returns to step 558 in Figure 7. If no further assistance is required, then CONSOLE.EXE 150 provides collected response data in step 614, and writes a BDR in step 616, as with steps 580 and 582 of Figure 7.

Referring to Figure 9, an exemplary and generic opening or initial display window or screen 300 is shown. The screen 300 is shown when a call is first received, and includes a greeting script portion 302 for the agent to read, and a portion 304 to enter the caller's name and address. The screen 300 also includes the customer's company name and address in a portion 306 and a notepad portion 308 on which the agent may record notes. Additionally, a window 310 displays messages from a call center supervisor to the agent.

The screen 300 also includes an order entry soft key 312 as well as status portion 314 which indicates the duration of the current call at the call center. After the agent enters the caller's name and address (if needed), CONSOLE.EXE 150 and GUI.EXE 152 display another screen under the appropriate call flow process being performed to receive additional caller data.

Referring to Figure 10, an exemplary order entry display window or screen 400 is shown. Details on an exemplary order entry call flow process are described below with respect to the screen 400. All such details under the process are performed, in general, by CONSOLE.EXE 150 and GUI.EXE 152.
The screen 400 includes a list of ordered items 402 forming approximately an upper section of the screen and having a group of editable fields including product code, and possibly non-editable fields. The product code field takes input from the agent either by entering the product code or by selecting an item in an item menu selection portion 404, forming a left-most middle section of the screen 400. The enter key on the keyboard 68 can validate the product code, concatenate the product description of the product code and move the cursor to a quantity field. Entering a value into the quantity field makes the item entry valid and updates the following fields: item price, shipping and handling, total cost,, and total in a portion 406 of the screen 400. The price of an item may vary depending upon the quantity ordered.

The item menu selection portion 404 provides a group of up to seven menus used to select items offered by the customer. The title of the menu and the list of items in the menu are data driven (specified in the customer profile). Selection of an item from the menu can cause a refresh of hierarchical menus underneath the current menu depending upon the items selected. Certain keystrokes can provide additional functionality. For example, pressing the Alt + tab key while on any menu or pressing the tab or enter key while on the last menu will cause an update of the current product code/description entry in the order entry portion 402 and move the cursor to the quantity field for input. The agent can scroll or move through lists of items in the menu portion 404 by pressing the arrow keys, while the tab, shift tab or the enter key selects the items and moves focus to the next menu in the hierarchy, in known fashion.

An additional info section 408 of the screen 400, forming a lower third portion of the screen, is reserved for customer defined controls. In general, controls and display information in the portion 408 are tied directly to the current product being offered. For example, a left hand portion of the additional info portion 408 of the screen 400 displays a box for controlling and collecting shipping options and a right hand portion for collecting shipping address. The shipping options portion of the portion 408 may provide a dropdown list of items to choose from, such as UPS first or second day delivery, Federal Express delivery, etc. Based on the selection of items from such box, the shipping and handling costs will vary. For example, shipping
options can include a flat charge for the entire order or a charge for each ship-to address. The charge for each ship-to address can be separate for each address, or a fixed additional cost for each additional ship-to address. A surcharge can be provided for multiple ship-to addresses, or the shipping and handling charge can vary based on a weight range stored in a weight table. Additionally, shipping and handling charges can be provided as a percentage of the entire order price, also provided in a lookup table, or a flat charge based on dollar value of the order.

A menu bar 410 provides the agent with controls over order entry processing, accessing help information, and displaying traditional agent console windows and edit utilities. Such functions are presented and organized under order, view, edit and help menu bar titles.

In general, data processed in the platform 105 is modeled in a tag-length-data format, in that each data record or field has three parts. First, a “tag” identifies the type of data, including a version of the type. Second, a “length” specifies the length of the entire record. Third, the actual data follows. Each aspect of a call flow process is specified in this format (i.e., what a question looks like to an agent, what is written to a BDR, product orders, etc.). Data records pulled from a customer profile are similarly in this format, as well as data collected from the agent at the agent console 118, and which is written to the NIDS server 120.

By specifying a generically-defined tag in each record, rather than defining an actual data type, the platform 105 is independent of operating systems and data type versions. If a certain component (e.g., PC containing the profile editor, agent console, ADS, boot server, etc.) changes, perhaps due to a change in the operating system, the data record still conforms to the interface specification. Each component maps a generically-defined tag to its own data type. Also, if a version of a data type in a certain component changes, it does not affect other components. In this manner, one can replace an OS/2 agent console 118 with a Windows agent console, or other operating system, since communication of data records in the exemplary embodiment is independent of the operating system or the computer that generated the data.
The agent console 118 preferably collects data in two general formats: data only and labeled data. Typically, an order entry call flow process provides only data in labeled data format, while survey and direct response can provide data in either format.

Under the data only format, data collected from callers is provided in predetermined positions, where such positions determine the field to which the data pertains. Customers preferably have the ability to define the field delimiters as well as record delimiters under the profile editor. In the exemplary embodiment, the first six fields of each record generated by the agent console 118 based on a call are automatically generated by CONSOLE.EXE 150, as shown below in Table 1. Subsequent fields are recorded by the agent in the order that they are stored in the customer profile, and thus in the order displayed to the agent on the screen. Consequently, each record includes the six initial fields plus the number of questions provided under the customer profile.

In the exemplary embodiment, all fields are stored in quotation marks, except for a status code and operator duration. Fields not collected by an operator are simply stored as a set of quotations without any intervening space or character indicating that no data was collected for that field. In the exemplary embodiment, a record delimiter is CR-LF. Table 1 below shows an exemplary data structure for data collected under a survey or direct response call flow process and exemplary data collected.
Table 1

<table>
<thead>
<tr>
<th>Field Number</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>BDR Sequence Number</td>
</tr>
<tr>
<td>2</td>
<td>Timestamp</td>
</tr>
<tr>
<td>3</td>
<td>Status</td>
</tr>
<tr>
<td>4</td>
<td>Duration</td>
</tr>
<tr>
<td>5</td>
<td>800 Number</td>
</tr>
<tr>
<td>6</td>
<td>Profile Start Date</td>
</tr>
<tr>
<td>7</td>
<td>1st Field</td>
</tr>
<tr>
<td>8</td>
<td>2nd Field</td>
</tr>
<tr>
<td>9</td>
<td>3rd Field</td>
</tr>
<tr>
<td>n</td>
<td>nth Field</td>
</tr>
</tbody>
</table>

Exemplary Data:

<table>
<thead>
<tr>
<th>Field Number</th>
<th>Field Name</th>
<th>Field Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>BDR Seq#</td>
<td>&quot;0380123412345678&quot;</td>
</tr>
<tr>
<td>2</td>
<td>Time/Date</td>
<td>&quot;Dec 17 1995 13:58&quot;</td>
</tr>
<tr>
<td>3</td>
<td>Status</td>
<td>101</td>
</tr>
<tr>
<td>4</td>
<td>Duration, Sec</td>
<td>231</td>
</tr>
<tr>
<td>5</td>
<td>800Numer</td>
<td>&quot;8005551212&quot;</td>
</tr>
<tr>
<td>6</td>
<td>Start Date</td>
<td>&quot;1201950000&quot;</td>
</tr>
<tr>
<td>7</td>
<td>Prior Order?</td>
<td>&quot;No&quot;</td>
</tr>
<tr>
<td>8</td>
<td>Contact</td>
<td>&quot;Via Friend&quot;</td>
</tr>
<tr>
<td>9</td>
<td>Day Phone</td>
<td>&quot;5552221111&quot;</td>
</tr>
<tr>
<td>10</td>
<td>Phone Extension</td>
<td>&quot;x2256&quot;</td>
</tr>
<tr>
<td>Field Number</td>
<td>Field Name</td>
<td>Field Example</td>
</tr>
<tr>
<td>--------------</td>
<td>------------</td>
<td>---------------</td>
</tr>
<tr>
<td>11</td>
<td>First Name</td>
<td>&quot;Jon&quot;</td>
</tr>
<tr>
<td>12</td>
<td>Last Name</td>
<td>&quot;Doe&quot;</td>
</tr>
<tr>
<td>13</td>
<td>Addr1</td>
<td>&quot;1260 3rd Street&quot;</td>
</tr>
<tr>
<td>14</td>
<td>Addr2</td>
<td>&quot;Suite 120&quot;</td>
</tr>
<tr>
<td>15</td>
<td>City</td>
<td>&quot;Iowa City&quot;</td>
</tr>
<tr>
<td>16</td>
<td>State</td>
<td>&quot;IA&quot;</td>
</tr>
<tr>
<td>17</td>
<td>Zip</td>
<td>&quot;52246&quot;</td>
</tr>
<tr>
<td>18</td>
<td>Mailing</td>
<td>&quot;Yes&quot;</td>
</tr>
</tbody>
</table>

Exemplary data output under the data only format for two consecutive records is presented below. The exemplary data presented below is listed as if the DOS command TYPE<filename> was executed on the data, so that the data would word wrap and the entire contents displayed on a computer screen.


Under the labeled data format, a label identifies every field collected by an agent in the corresponding collected data record. Thus, the agent console 118 can handle variable amounts of fields and collected data for each received call, such as with an order entry call flow process. Additionally, the labeled data format provides a very generic version of "ASCII free form text." Under the exemplary embodiment, the customer can define values for three items in the resulting records: column separator, field separator and record delimiter.

Field labels define the first six fields automatically collected under CONSOLE.EXE 150, which each begin with a "@" symbol. Field labels for data
collected by an agent are defined in each customer profile and must have a unique value. All collected data can be written in quotes except status code, operator duration, order entry, quantity and order entry price.

Two options for data population of fields under order entry call flow processes are preferably provided: survey data repeated and single survey data. Such options are defined in the customer profile. Under the survey data repeated option, CONSOLE.EXE 150 repeats all survey questions collected by the agent and credit card data for every item purchased. Thus, a data record for every item purchased will include all data that was collected under the survey portion of the call along with the credit card data. For example, if three items are purchased during a single order entry call, then CONSOLE.EXE 150 produces three collected data records, each with the same six reserved system fields (the first six fields), survey information (e.g., customer name and address) and credit card data.

Under the single survey data format, CONSOLE.EXE 150 writes a single collected data record, even if multiple items are purchased. CONSOLE.EXE 150 can collect only fields associated with the product being sold. Only those fields associated with a given product are collected and written to output records. For example, if a product A requires two additional fields to be collected such as cookie flavor and greeting card, while product B requires three items such as cookie flavor, greeting card and personalization data, CONSOLE.EXE 150 only records two or three additional fields for products A and B, respectively.

Table 2 below shows the customer definable options under the labeled data format. Such options are defined in the customer's profile.
Table 2

<table>
<thead>
<tr>
<th>Delimiter Type</th>
<th>Function</th>
<th>Typical Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field</td>
<td>Separates fields inside of a collected data record</td>
<td><code>&lt;CR&gt;</code>&lt;LF&gt;</td>
</tr>
<tr>
<td>Record</td>
<td>Separates individual records</td>
<td>@newrec or <code>&lt;CR&gt;</code>&lt;LF&gt;</td>
</tr>
<tr>
<td>Column</td>
<td>Separates the field label from the field data</td>
<td>Comma and a tab</td>
</tr>
</tbody>
</table>

Table 3 below shows an exemplary data structure for a record under the labeled data format having both the initial six system defined labels, and subsequent customer defined labels (as defined in the customer profile).

<table>
<thead>
<tr>
<th>Field Number</th>
<th>Contents</th>
<th>Field Label Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>BDR Sequence Number</td>
<td>@BDRSeqNum</td>
</tr>
<tr>
<td>2</td>
<td>Timestamp</td>
<td>@TimeDate</td>
</tr>
<tr>
<td>3</td>
<td>Status</td>
<td>@Status</td>
</tr>
<tr>
<td>4</td>
<td>Duration</td>
<td>@Duration</td>
</tr>
<tr>
<td>5</td>
<td>800 Number</td>
<td>@800Num</td>
</tr>
<tr>
<td>6</td>
<td>Profile Start Date</td>
<td>@StartDate</td>
</tr>
<tr>
<td>7</td>
<td>1st Field</td>
<td>&lt;from profile&gt;</td>
</tr>
<tr>
<td>8</td>
<td>2nd Field</td>
<td>&lt;from profile&gt;</td>
</tr>
<tr>
<td>9</td>
<td>3rd Field</td>
<td>&lt;from profile&gt;</td>
</tr>
<tr>
<td>n</td>
<td>n&lt;sup&gt;th&lt;/sup&gt;</td>
<td>&lt;from profile&gt;</td>
</tr>
</tbody>
</table>

Presented below is an exemplary record under the labeled data format, employing the following three customer defined options: column delimiter of “comma plus tab,” field delimiter of “CR-FL” and record delimiter of “<pagebreak>”.
In the exemplary embodiment, status codes are also written to collected data records to document the manner in which calls are terminated. Such status codes can be employed by the customers to filter data collected by the agents. Many calls may be terminated and recorded as “successful,” while various other calls may be unsuccessful for a variety of reasons. Table 4 below shows exemplary decimal and hexadecimal values for status codes, and corresponding descriptions for each code.

<table>
<thead>
<tr>
<th>Decimal Value</th>
<th>Hex Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>257</td>
<td>101</td>
<td>Success, valid call or order, all required fields collected and the operator has pressed the “DONE” soft key.</td>
</tr>
<tr>
<td>5376</td>
<td>1500</td>
<td>Caller disconnects prior to the “DONE” key being displayed. The done key is presented to the operator only after all required fields have been collected.</td>
</tr>
<tr>
<td>5377</td>
<td>1501</td>
<td>Caller disconnects after the done key is displayed and the operator selected the ‘exit’ or ‘cancel’ key (Operator discretion that the call was not complete)</td>
</tr>
<tr>
<td>5378</td>
<td>1502</td>
<td>The caller disconnects on an order prior to the credit card number being launched for validation (prior to the complete order key being selected)</td>
</tr>
<tr>
<td>5379</td>
<td>1503</td>
<td>The operator disconnects (flushes) the call.</td>
</tr>
<tr>
<td>5380</td>
<td>1504</td>
<td>The ‘Cancel Order’ key was pressed by the operator</td>
</tr>
<tr>
<td>5381</td>
<td>1505</td>
<td>The call was transferred to customer service.</td>
</tr>
</tbody>
</table>

In an alternative embodiment, CONSOLE.EXE 150 and other logical components of the agent console 118 shown in Figure 5 can be implemented in the audio response unit ARU 108. As is known, an ARU is commonly used to provide automated operator services. The call flow processes can run on an ARU by receiving calls, prompting the caller for input, and recording the caller’s spoken and DTMF digit entry. Spoken responses can be later transcribed to text or converted to text using voice recognition technology, while DTMF signals are automatically converted to answers or
information that they represent. All data is placed in the results file for the service provider or customer.

Under an exemplary alternative survey call flow process, the caller initially places a 1-800 call that is received by the ARU 108, which in turn retrieves the appropriate customer profile and executes the call flow process. The ARU 108 provides a greeting to the caller and requests the caller leave his or her name and complete mailing address at the beep, and the ARU provides an audible beep tone and pauses to receive and record the caller’s spoken name and address. The ARU 108 may also request that the caller provide additional information under questions specified in the profile. Numeric input can be provided by DTMF input. The ARU 108 preferably echoes entered or spoken answers from the caller and requests the caller to press a key if the entered information is correct, otherwise to repronounce or reenter the information. The ARU 108 may request the caller to answer questions by pressing certain keys for yes and no. The ARU 108 also provides an exit or final message to the caller following the questions. Thereafter, the ARU 108 transcribes calls automatically, or by an agent listening to the recorded names and addresses of callers, which are matched to DTMF entry to complete a record. Customer addresses can be verified by an address verification routine that adds zip codes.

Under any of the above-described embodiments, the exemplary embodiment can provide a prerecorded call announcement or message for callers after dialing into the platform 10. The announcement serves a dual purpose of: (1) advising callers waiting in queue that their call will be answered by the next available operator (or ARU), or (2) to advise callers about various marketing promotions or products and services offered by the customer. Additionally, the announcement can provide a standard greeting for the customer which pronounces the customer’s name as well as other information as desired by the customer and/or provides other messages which the customer desires to present to callers.

Although specific embodiments of, and examples for, the present invention are described herein for illustrative purposes, various equivalent modifications can be made without departing from the spirit and scope of the invention,
as will be recognized by those skilled in the relevant art. The teachings provided herein of the present invention can be applied to other call processing systems, not necessarily the exemplary survey and order entry process systems described above. For example, the present invention can be applied to systems for providing electronic input from surveys provided to client/prospects in an Internet-driven survey. Additionally, the process can be employed to provide technical support or other services for customers and their clients/prospects.

The teachings provided herein of the present invention can be applied to other call processing, data gathering, and record producing systems, not necessarily the exemplary telecommunications system described above. Aspects and embodiments of the present invention can be combined to provide yet further systems and processes.

All U.S. patents and applications cited above are incorporated by reference herein.

These and other changes can be made to the invention in light of the above detailed description. In general, in the following claims, the terms used should not be construed to limit the invention to the specific embodiments disclosed in the specification and the claims, but should be construed to include all call processing and data collection systems that operate under the claims to provide a method for gathering data and producing records therefrom. Accordingly, the invention is not limited by the disclosure, but instead its scope is to be determined entirely by the following claims.
CLAIMS

1. A computer-implemented method for use in a telecommunications system comprising the steps of:
   creating a plurality of profiles of call flow data specified by users;
   storing the plurality of profiles;
   retrieving one of the plurality of profiles;
   defining a call flow process based on the retrieved profile and a generic executable routine, wherein the call flow process provides questions for a call receiver handling telephone calls; and
   storing data input by the call receiver based on answers to the questions.

2. The method of claim 1 wherein the step of storing the plurality of profiles includes the step of storing the plurality of profiles based on telephone numbers of the users, and wherein the step of retrieving includes the step of retrieving one of the plurality of profiles after receiving a telephone call to a corresponding one of the plurality of telephone numbers.

3. The method of claim 1 wherein the call receiver is a call center agent, wherein the step of defining includes the step of displaying standard data for the agent based on the executable routine, and displaying specific data specified by the retrieved profile, wherein the standard data includes a layout of a display window, and wherein the specific data includes the questions.

4. The method of claim 1 wherein the call receiver is a call center agent, wherein the step of defining includes the step of providing the executable routine, and wherein the executable routine includes a user interface portion for defining data displayable to the agent, and a data manipulatable portion that receives and employs data in the profile.

5. The method of claim 1 wherein the step of storing data includes the steps of:
storing records of each call processed by the call receiver corresponding to the retrieved profile; and
electronically transferring the records to a storage location.

6. The method of claim 1, further comprising the step of creating a billing record for each call processed by the call receiver corresponding to the retrieved profile.

7. The method of claim 1 wherein the step of defining includes the step of generating either a caller survey call flow process or a order entry call flow process based on the retrieved profile.

8. The method of claim 1 wherein the call receiver is an automated response unit, and wherein the step of storing data includes the step of recording spoken responses to the questions.

9. A telecommunication system for receiving and processing calls comprising:
a server for storing a plurality of profiles specified by users, wherein each profile specifies details of a specific call flow process; and
at least one call receiving station coupled to the server and coupled to receive incoming telephone calls, wherein the call receiving station retrieves one of the plurality of profiles and defines a call flow process based on the retrieved profile and an executable routine, wherein the defined call flow process provides questions for an incoming telephone call, and wherein the call receiving station collects answers to the questions.

10. The system of claim 9, further comprising an automated call distributor coupled to first and second call receiving stations, wherein the first call receiving station is an automated response unit and the second call receiving station is a computer workstation.
11. The system of claim 9 wherein the plurality of profiles are retrievable from the server based on telephone numbers of the users, and wherein the call receiving station retrieves one of the plurality of profiles after receiving a telephone call to a corresponding one of the telephone numbers.

12. The system of claim 9 wherein the call receiving station is a computer workstation for a call center agent, wherein the workstation displays standard data for the agent based on the executable routine, and displays specific data specified by the retrieved profile.

13. The system of claim 9 wherein the call receiving station is a computer workstation for a call center agent, and wherein the executable routine includes a user interface portion for defining data displayable to the agent, and a data manipulatable portion that receives and employs data in the profile.

14. The system of claim 9, further comprising a network server coupled to the call receiving station, and wherein the call receiving station creates records of each processed call corresponding to the retrieved profile, and electronically transfers the records to the network server.

15. The system of claim 9 wherein the call receiving station generates either a caller survey call flow process or an order entry call flow process based on the retrieved profile.

16. A computer-implemented method for use in a telecommunications system comprising the steps of:

   retrieving one of a plurality of stored profiles, each profile containing call flow data specified by users; and
defining a call flow process based on the retrieved profile and an executable routine, wherein the call flow process provides questions for a call receiver that services telephone calls.

17. The method of claim 16, further comprising the step of storing data input by the call receiver based on answers to the questions.

18. The method of claim 16 wherein the step of retrieving includes the step of retrieving one of the plurality of profiles after receiving a telephone call to a corresponding one of a plurality of telephone numbers.

19. The method of claim 16 wherein the step of retrieving includes the step of retrieving a property record indicating which one of the plurality of profiles to retrieve based on the corresponding one of a plurality of telephone numbers, wherein at least one profile corresponds to two or more telephone numbers.

20. The method of claim 16 wherein the step of retrieving includes the step of retrieving one of the plurality of profiles, wherein the retrieved profile specifies a format in which data provided in response to the questions is to be formatted.

21. The method of claim 16 wherein the call receiver is a call center agent, wherein the step of defining includes the step of displaying standard data for the agent based on the executable routine, and displaying specific data specified by the retrieved profile, wherein the standard data includes a layout of a display window, and wherein the specific data includes the questions.

22. The method of claim 16 wherein the call receiver is a call center agent, wherein the step of defining includes the step of providing the executable routine, and wherein the executable routine includes a user interface portion for defining data displayable to the agent, and a data manipulatable portion that incorporates data in the profile.
23. The method of claim 16, further comprising the steps of:
    storing records of each call processed by the call receiver corresponding to the
    retrieved profile; and
    electronically transferring the records to another location.

24. The method of claim 16, further comprising the step of creating a
    billing record for each call processed by the call receiver corresponding to the retrieved
    profile.

25. The method of claim 16 wherein the step of defining includes the step
    of generating either a caller survey call flow process or an order entry call flow process based
    on the retrieved profile.

26. The method of claim 16 wherein the call receiver is an automated
    response unit, and wherein the method further includes the step of storing includes the step of
    recording spoken responses to the questions.

27. A computer-readable medium storing computer-executable steps under
    a method, the method comprising the steps of:
    retrieving one of a plurality of stored profiles, each profile containing call flow
    data specified by users; and
    defining a call flow process based on the retrieved profile and a generic
    executable routine, wherein the call flow process provides questions for a call receiver
    handling telephone calls.

28. The method of claim 27, further comprising the step of storing data
    input by the call receiver based on answers to the questions.
29. The method of claim 27 wherein the step of retrieving includes the step of retrieving one of the plurality of profiles after receiving a telephone call to a corresponding one of a plurality of telephone numbers, wherein the plurality of profiles are stored based on the plurality of telephone numbers.

30. The method of claim 27 wherein the call receiver is a call center agent, wherein the step of defining includes the step of displaying standard data for the agent based on the executable routine, and displaying specific data specified by the retrieved profile, wherein the standard data includes a layout of a display window, and wherein the specific data includes the questions.

31. The method of claim 27 wherein the step of defining includes the step of generating either a caller survey call flow process or an order entry call flow process based on the retrieved profile.

32. The method of claim 27 wherein the call receiver is an automated response unit, and wherein the method further includes the step of storing includes the step of recording spoken responses to the questions.

33. The method of claim 27 wherein the step of retrieving includes the step of retrieving one of the plurality of profiles, wherein the retrieved profile specifies a format in which data provided in response to the questions is to be formatted.
Fig. 3
Fig. 5
6/12

Create customer profile

Upload customer profile to ADS

Download customer profiles from ADS to Boot Servers

Agent Console retrieves customer profiles from Boot Server

Agent Console performs call flow application to handle call

Agent Console writes data collected from caller to NIDS Server

Agent Console writes BDR from call to NIDS Server

Fig. 6
Fig. 7A

AnswerNet Live
High Level Flow

550

Read profile data

552

Populate Relevant Fields in Display Window

554

OE?

Yes

Display and enable OE key

556

No

A
Fig. 8

600

602

604

606

612

614

616

Return to calling function

Further assistance required?

Write collected response data

Write BDR

Complete call to CS #

Inform the caller of the business hours for the CS #, give them the CS #, and offer to help them further

Is the current time within the range of business hours for this CSP?

Write collected response data

Write BDR
"Hello, welcome to ABC Corp. May I have your name, please?"

Company name and address:
ABC Corp.
P.O. Box 0000
New York, NY 99999

Caller information:
Name:
Address 1:
Address 2:
City:
State:
Zip:

Call Duration

Order Entry

Fig. 9
Andrew's Infant Shop OE Preica

Order Edit View Help

Product Code: DABGPR

Name: Diaper Bag - Primary

Item Menu Selection

<table>
<thead>
<tr>
<th>Baby Category</th>
<th>Diaper Accessories</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baby Item Name</td>
<td>Diaper Bag</td>
<td>X</td>
</tr>
<tr>
<td>Color</td>
<td>Primary</td>
<td>X</td>
</tr>
</tbody>
</table>

Item Price: 20.00
S&H Surcharge: 10.00
X Quantity: 1
Total: 30.00
Total Order:

Customer Options

- Shipping Destination
- Shipping Selections
- Additional Comments

Shipping Destination

Last Name: SAMPSON
Address Line 1: 500 SOUTH 2ND AVE
Address Line 2:
City: CEDAR RAPIDS
State: IA
Zip Code:

Item 1 of 1

Fig. 10
INTERNATIONAL SEARCH REPORT

A. CLASSIFICATION OF SUBJECT MATTER

IPC 6 H04M3/50

According to international Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 6 H04M

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>EP 0 545 226 A (ROLM CO) 9 June 1993 see page 2, line 30 - line 48 see page 4, line 35 - page 5, line 5</td>
<td>1-33</td>
</tr>
<tr>
<td>X</td>
<td>EP 0 501 189 A (IBM) 2 September 1992 see page 19, line 8 - line 41</td>
<td>1-33</td>
</tr>
<tr>
<td>X</td>
<td>WO 96 05685 A (DIGITAL SYSTEMS INT) 22 February 1996 see page 1, line 32 - page 2, line 22 see page 26, line 30 - page 27, line 5 see page 3, line 15 - line 33</td>
<td>1,3-7,9, 10, 12-17, 21-25, 27,28, 30,31</td>
</tr>
</tbody>
</table>

Further documents are listed in the continuation of box C.

Patent family members are listed in annex.

\* Special categories of cited documents:
\*A* document defining the general state of the art which is not considered to be of particular relevance
\*E* earlier document but published on or after the international filing date
\*L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
\*O* document referring to an oral disclosure, use, exhibition or other means
\*P* document published prior to the international filing date but later than the priority date claimed

Date of the actual completion of the international search

3 September 1998

Date of mailing of the international search report

09/09/1998

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV RIJSWIJK Tel: (+31-70) 340-2040, Tx: 31 651 epo nl, Fax: (+31-70) 340-3016

Authorized officer

Vandevenne, M
<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>WO 92 09164 A (UNIFI COMMUNICATIONS CORP) 29 May 1992</td>
<td>1-4,7, 9-13,15, 16,18, 19,21, 22,25,27</td>
</tr>
<tr>
<td></td>
<td>see page 1, line 27 - page 2, line 7</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td>GB 2 273 853 A (ROCKWELL INTERNATIONAL CORP) 29 June 1994</td>
<td>1-4,7, 9-13,15, 16,18, 19,21, 22,25,27</td>
</tr>
<tr>
<td></td>
<td>see page 4, line 3 - line 18</td>
<td></td>
</tr>
<tr>
<td></td>
<td>see page 9, line 35 - page 11, line 33</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td>US 5 001 710 A (GAWRYS GEORGE W ET AL) 19 March 1991</td>
<td>1,2,9, 11,16, 18,19, 27,29</td>
</tr>
<tr>
<td></td>
<td>see column 3, line 42 - column 4, line 38</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td>US 4 852 149 A (ZWICK NICHOLAS ET AL) 25 July 1989</td>
<td>1,2,9, 11,16, 18,19, 27,29</td>
</tr>
<tr>
<td></td>
<td>see column 6, line 12 - column 7, line 10</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>HARVEY D E ET AL: &quot;CALL CENTER SOLUTIONS&quot; AT &amp; T TECHNICAL JOURNAL, vol. 70, no. 5, 1 September 1991, pages 36-44, XP000244602</td>
<td></td>
</tr>
<tr>
<td></td>
<td>see page 40, left-hand column, line 17 - right-hand column, line 17</td>
<td></td>
</tr>
<tr>
<td>Patent document cited in search report</td>
<td>Publication date</td>
<td>Patent family member(s)</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>-----------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>EP 0545226 A</td>
<td>09-06-1993</td>
<td>NONE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JP 1938744 C</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JP 4353957 A</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JP 6070782 B</td>
</tr>
<tr>
<td>WO 9605685 A</td>
<td>22-02-1996</td>
<td>CA 2197456 A</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CA 2095916 A</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EP 0670094 A</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 5299260 A</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JP 6225022 A</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 5544232 A</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DE 69022354 D</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DE 69022354 T</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EP 0425163 A</td>
</tr>
<tr>
<td>US 4852149 A</td>
<td>25-07-1989</td>
<td>NONE</td>
</tr>
</tbody>
</table>